



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΥΠΟΥΡΓΕΙΟ
ΠΑΙΔΕΙΑΣ, ΕΡΕΥΝΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ

ΓΕΝΙΚΗ ΔΙΕΥΘΥΝΣΗ ΣΠΟΥΔΩΝ
Π/ΘΜΙΑΣ ΚΑΙ Δ/ΘΜΙΑΣ ΕΚΠΑΙΔΕΥΣΗΣ
ΔΙΕΥΘΥΝΣΗ ΣΠΟΥΔΩΝ, ΠΡΟΓΡΑΜΜΑΤΩΝ
ΚΑΙ ΟΡΓΑΝΩΣΗΣ Δ/ΘΜΙΑΣ ΕΚΠ/ΣΗΣ
ΤΜΗΜΑ Α΄

Ταχ. Δ/ση: Ανδρέα Παπανδρέου 37
Τ.Κ. – Πόλη: 15180 Μαρούσι
Ιστοσελίδα: www.minedu.gov.gr
Πληροφορίες: Αν. Πασχαλίδου
Τηλέφωνο: 210-3443422

Βαθμός Ασφαλείας:
Να διατηρηθεί μέχρι:
Βαθ. Προτεραιότητας:

Αθήνα, 08-12-2015
Αρ. Πρωτ. 199465 /Δ2

ΠΡΟΣ:

- Περιφερειακές Δ/νσεις Εκπ/σης
- Σχολ. Συμβούλους Δ.Ε. (μέσω των Περιφερειακών Δ/νσεων Εκπ/σης)
- Δ/νσεις Δ/θμιας Εκπ/σης
- Γενικά Λύκεια (μέσω των Δ/νσεων Δ/θμιας Εκπ/σης)

ΚΟΙΝ.:

Ινστιτούτο Εκπαιδευτικής
Πολιτικής
Αν. Τσόχα 36
11521 Αθήνα

ΘΕΜΑ: Οδηγίες για τη διδασκαλία του μαθήματος «Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον» της Γ΄ τάξης Ημερήσιου Γενικού Λυκείου για το σχολ. έτος 2015 – 2016

Μετά από σχετική εισήγηση του Ινστιτούτου Εκπαιδευτικής Πολιτικής (πράξη 57/18-11-2015 του Δ.Σ) σας αποστέλλουμε τις παρακάτω οδηγίες για τη διδασκαλία του μαθήματος «Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον» Ομάδας Προσανατολισμού Θετικών Σπουδών και Ομάδας Προσανατολισμού Σπουδών Οικονομίας και Πληροφορικής της Γ΄ τάξης Ημερήσιου Γενικού Λυκείου για το σχολικό έτος 2015- 2016.

ΓΕΝΙΚΕΣ ΟΔΗΓΙΕΣ ΔΙΔΑΣΚΑΛΙΑΣ

Παρατηρήσεις:

- Οι Αλγόριθμοι να υλοποιούνται σε αμιγώς προγραμματιστικό περιβάλλον και συγκεκριμένα αυτό της ΓΛΩΣΣΑΣ.
- Να γίνει επισκόπηση της έννοιας του αλγορίθμου, των χαρακτηριστικών του και των τρόπων αναπαράστασής του, καθώς και εισαγωγή στα χαρακτηριστικά των γλωσσών προγραμματισμού και ειδικά της ΓΛΩΣΣΑΣ.
- Οι βασικές αλγοριθμικές δομές του κεφαλαίου 2 (ακολουθίας, επιλογής και επανάληψης) να διδαχθούν συνοπτικά και παράλληλα με το κεφάλαιο 7 και 8, στην κατεύθυνση της κάλυψης τυχόν γνωσιακών κενών από την προηγούμενη τάξη, με τις ασκήσεις να υλοποιούνται σε ΓΛΩΣΣΑ.
- Στο κεφάλαιο 3:
 - Να προστεθούν ασκήσεις στη *στοίβα* και *ουρά*, που επίσης θα υλοποιηθούν σε ΓΛΩΣΣΑ και με την πρόσθεση της θεωρίας της ενότητας 3.9.
 - Οι δομές της ενότητας 3.9 (*λίστες*, *δένδρα*, *γράφοι*) να διδαχθούν αποκλειστικά ως θεωρία και στο επίπεδο ανάλυσης του βιβλίου.

- ο Οι πίνακες να διδαχθούν παράλληλα με το κεφάλαιο 9, με τις ασκήσεις να υλοποιούνται σε ΓΛΩΣΣΑ.
- ο Εισάγονται νέοι αλγόριθμοι αναζήτησης και ταξινόμησης σε πίνακες (ως ασκήσεις).
- Στο κεφάλαιο 5 να διδαχθούν οι ενότητες 5.1 (επίδοση αλγορίθμων) και 5.3 (πολυπλοκότητα αλγορίθμων). Η έννοια της επίδοσης να εξεταστεί με αναφορά στους αλγορίθμους αναζήτησης και ταξινόμησης. Η πολυπλοκότητα αλγορίθμων να διδαχθεί θεωρητικά με παραδείγματα και σε σύνδεση με την επίδοση, χωρίς οι μαθητές να εμπλακούν σε ασκήσεις υπολογισμού της τάξης O ενός αλγορίθμου.
- Από το κεφάλαιο 6 να διδαχθούν οι ενότητες 6.3, 6.4 και 6.7. Η παράγραφος 6.3 διδάσκεται στην αρχή του κεφαλαίου 7 ενώ οι παράγραφοι 6.4 και 6.7 στο τέλος του κεφαλαίου 7.
- Στα κεφάλαια 7, 8 και 9 δεν επέρχεται ουδεμία μεταβολή.
- Στο κεφάλαιο 10 προστίθεται η ενότητα 10.6 (εμβέλεια μεταβλητών - σταθερών).

Οι ανωτέρω παρατηρήσεις έχουν λάβει υπόψη τη διδασκαλία των Αλγορίθμων στη Β' Λυκείου, όπου οι μαθητές έχουν διδαχθεί τη γραφή αλγόριθμου σε ψευδογλώσσα και την αναπαράσταση αλγορίθμων με διαγραμματικές τεχνικές. Κατά τη διδασκαλία του μαθήματος στη Γ' Λυκείου, οι μαθητές εξοικειώνονται με την υλοποίηση αλγορίθμων σε αμιγώς προγραμματιστικό περιβάλλον και συγκεκριμένα αυτό της ΓΛΩΣΣΑΣ. Η ψευδογλώσσα και τα διαγράμματα ροής θεωρούνται ήδη γνωστά και στη Γ' Λυκείου καλύπτονται μόνο πιθανά κενά από τη διδασκαλία τους στη Β' Λυκείου.

Η εισαγωγή νέων αλγορίθμων αναζήτησης και ταξινόμησης σε πίνακες, αφορά στη διδασκαλία ασκήσεων στις οποίες να περιγράφεται ο αλγόριθμος αναζήτησης ή ταξινόμησης και να ζητείται από τους μαθητές η υλοποίηση του σε πρόγραμμα.

Η διδασκαλία του κεφαλαίου 5 (Ανάλυση Αλγορίθμων) αποσκοπεί στο να γνωρίσουν και να κατανοήσουν, οι μαθητές, απλά θέματα σχετικά με την πολυπλοκότητα, την επίδοση και την αποδοτικότητα αλγορίθμων, που επιλύουν το ίδιο πρόβλημα. Ενδεικτικές ασκήσεις αναφέρονται στις παραγράφους 5.1.1 και 5.1.3. του βιβλίου μαθητή και στο τετράδιο μαθητή. Επίσης, κατά τη διδασκαλία, να ενθαρρύνονται οι μαθητές να διατυπώνουν για το ίδιο πρόβλημα εναλλακτικές προγραμματιστικές λύσεις, όπως και να συγκρίνουν μεταξύ τους δοθείσες προγραμματιστικές λύσεις, μέσα στο πλαίσιο που ορίζεται από τα σχολικά εγχειρίδια και τις οδηγίες διδασκαλίας. Σε κάθε περίπτωση, αυτό πρέπει να ζητείται ευκρινώς στη διατύπωση της άσκησης και να μην θεωρείται αυτονόητο, καθώς επίσης και να προσδιορίζονται τα θέματα που αφορούν στη βαθμολόγηση της.

Ενδεικτικός Χρονοπρογραμματισμός και Ροή της Διδασκαλίας.

Ο ενδεικτικός προγραμματισμός και η προτεινόμενη ροή της διδασκαλίας αναπτύσσονται στον παρακάτω πίνακα.

A/A	Ενότητες	Περιγραφή	Ώρες
1	Εισαγωγικό μάθημα	Σύνδεση με το μάθημα της Β' ΓΕΛ Επανάληψη εννοιών: Τι είναι αλγόριθμος. Περιγραφή και αναπαράσταση αλγορίθμων.	2
2	2.1, 2.3		

3	6.3, 7.1, 7.2, 7.3, 7.4	Φυσικές και τεχνητές γλώσσες. Το αλφάβητο της ΓΛΩΣΣΑΣ, Τύποι Δεδομένων. Σταθερές, Μεταβλητές (με ΑΣΚΗΣΕΙΣ)	2
4	7.5, 7.6, 7.7	Αριθμητικοί τελεστές, Συναρτήσεις, Αριθμητικές Εκφράσεις (με ΑΣΚΗΣΕΙΣ)	1
5	7.8, 2.4.1, 7.9, 7.10.	Εντολή εκχώρησης, Εντολές εισόδου – εξόδου, Δομή προβλήματος. Δομή ακολουθίας	1
6	6.4	Τεχνικές Σχεδίασης προγραμμάτων	1
7	6.7	Προγραμματιστικά περιβάλλοντα	1
8	2.4.2, 2.4.3 2.4.4	Δομή επιλογής, Διαδικασίες πολλαπλών επιλογών, εμφωλευμένες διαδικασίες	2
9	8.1, 8.1.1	Εντολές επιλογής	1
10	2.4.5, 8.2, 8.2.1	Δομή επανάληψης. Εντολές επανάληψης, Εντολή ΟΣΟ...ΕΠΑΝΑΛΑΒΕ	1
11	8.2.2	Εντολή ΜΕΧΡΙΣ...ΟΤΟΥ	1
12	8.2.3	Εντολή ΓΙΑ...ΑΠΟ...ΜΕΧΡΙ	1
13		Μετατροπές από μία δομή επανάληψης σε άλλη	2
14		Γενικές Ασκήσεις εμπέδωσης μέχρι και την Δομή Επανάληψης	2
15	3.2	Αλγόριθμοι + Δομές Δεδομένων = Προγράμματα	1
16	3.3	Πίνακες	1
17	9.1	Μονοδιάστατοι πίνακες	1
18	3.6	Αναζήτηση	1
19	3.7	Ταξινόμηση	2
20	3.4	Στοιβά	1
21	3.5	Ουρά	1
22	3.9	Άλλες δομές δεδομένων	1
23	5.1 (5.1.1, 5.1.2, 5.1.3, 5.1.4)	Επίδοση αλγορίθμων	1
24	5.3	Πολυπλοκότητα Αλγορίθμων	3
25	9.2, 9.4	Πότε χρησιμοποιούνται πίνακες, Τυπικές επεξεργασίες πινάκων,	1
26	9.3	Πολυδιάστατοι πίνακες	3

27		Γενικές Ασκήσεις εμπέδωσης με πίνακες	2
28	10.1, 10.2, 10.3, 10.4	Τμηματικός προγραμματισμός, χαρακτηριστικά των υποπρογραμμάτων. Πλεονεκτήματα του τμηματικού προγραμματισμού, Παράμετροι	1
29	10.5	Διαδικασίες και συναρτήσεις	2
30	10.6	Εμβέλεια μεταβλητών - σταθερών	1
31		Γενικές Ασκήσεις εμπέδωσης με διαδικασίες και συναρτήσεις	5
		ΣΥΝΟΛΟ ΩΡΩΝ	46

Ο παραπάνω προγραμματισμός και η ροή της διδασκαλίας προτείνονται ενδεικτικά. Οι διδάσκοντες, ανάλογα με τις ανάγκες των τμημάτων τους, να προθούν σε εκείνες τις αλλαγές που επιβάλλονται για την ορθότερη επίτευξη των στόχων του μαθήματος.

ΠΡΟΣΟΧΗ !

Ορισμένοι ορισμοί στο βιβλίο της Β' ΓΕΛ, «Εισαγωγή στις Αρχές της Επιστήμης των Η/Υ» (ΕΑΕΗΥ), είναι ελαφρώς διαφορετικά διατυπωμένοι από τους αντίστοιχους του Βιβλίου Μαθητή της Γ' ΓΕΛ «**Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον**» (ΑΕΠΠ). Σε κάθε περίπτωση αυτοί θα διδαχθούν σύμφωνα με το βιβλίο του μαθήματος ΑΕΠΠ της Γ' ΓΕΛ.

Οι μαθητές πρέπει να διατυπώνουν τις λύσεις των ασκήσεων των εξετάσεων σε ΓΛΩΣΣΑ εκτός και αν αναφέρεται στην εκφώνηση διαφορετική μορφή αναπαράστασης του αλγορίθμου.

Ασκήσεις ή παραδείγματα του βιβλίου μαθητή ή του τετραδίου μαθητή, που χρησιμοποιούν την ΕΠΙΛΕΞΕ, η οποία έχει εξαιρεθεί, θα αντιμετωπίζονται με τη χρήση άλλης δομής επιλογής.

Οδηγίες Διδασκαλίας σύμφωνα με την προτεινόμενη ροή του μαθήματος

1. Εισαγωγικό Μάθημα

Ο διδάσκων, αναφέρεται συνοπτικά (τίτλοι κεφαλαίων, υποενότητες) στο περιεχόμενο της Ενότητας 2. **ΘΕΜΑΤΑ ΘΕΩΡΗΤΙΚΗΣ ΕΠΙΣΤΗΜΗΣ ΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ**, του βιβλίου «Εισαγωγή στις Αρχές της Επιστήμης των Η/Υ» της Β' ΓΕΛ. Συγκεκριμένα υπενθυμίζει ότι:

1. Οι μαθητές διδάχθηκαν την έννοια του προβλήματος και τις κατηγορίες προβλημάτων.
2. Ορίστηκε ο αλγόριθμος, και αναδείχθηκαν τα χαρακτηριστικά του αλλά και στοιχεία από την ανάλυση αλγορίθμου.
3. Γνώρισαν οι μαθητές βασικούς τύπους αλγορίθμων αλλά και τρόπους αναπαράστασής τους.
4. Χρησιμοποιήθηκαν εντολές και δομές αλγορίθμου με χρήση ψευδογλώσσας.

5. Περιγράφηκαν βασικές αλγοριθμικές λειτουργίες σε δομές δεδομένων.
6. Έγινε αναφορά σε γλώσσες προγραμματισμού και «Προγραμματιστικά Υποδείγματα».

Με βάση αυτό το υπόβαθρο, στην τρέχουσα τάξη, οι μαθητές θα αποκτήσουν στέρεη γνώση των σχετικών εννοιών, υλοποιώντας απλές Εφαρμογές σε ένα Εκπαιδευτικό Προγραμματιστικό Περιβάλλον.

Διάρκεια: Μία διδακτική ώρα

2. Ενότητες 2.1, 2.3

Στόχοι της ενότητας αυτής είναι, οι μαθητές να είναι σε θέση να:

- Δίνουν τον ορισμό του αλγόριθμου.
- Περιγράφουν τα κριτήρια που πρέπει να ικανοποιεί ένας αλγόριθμος.
- Αναφέρουν θεματικές περιοχές με τις οποίες συνδέονται οι αλγόριθμοι.
- Περιγράφουν τις βασικές τεχνικές στην αναπαράσταση αλγόριθμου.
- Χρησιμοποιούν τα βασικά σχήματα διαγράμματος ροής.

Οι έννοιες που εμπεριέχονται στις ενότητες 2.1 και 2.3 έχουν διδαχθεί στις ενότητες 2.2.1, 2.2.2 & 2.2.5 του μαθήματος «Εισαγωγή στις Αρχές της Επιστήμης των Η/Υ» της Β' ΓΕΛ. Μεταξύ των δύο βιβλίων δεν υπάρχουν αντιθέσεις σε σχέσεις με τους ορισμούς ή τη χρήση των εννοιών. Το βιβλίο της Β' ΓΕΛ εισάγει απ' ευθείας τους μαθητές στην κωδικοποίηση των αλγορίθμων μέσω ψευδογλώσσας.

Προτεινόμενη διδακτική προσέγγιση:

Μέσω καταιγισμού ιδεών και αναζήτησης, εργαζόμενοι οι μαθητές σε ομάδες, να επαναλάβουν συνοπτικά το κεφάλαιο, αφού οι έννοιες αυτές αναφέρθηκαν στην Β' Τάξη. Προτείνεται οι μαθητές να εμβαθύνουν στις έννοιες Αλγόριθμος, στα χαρακτηριστικά του, τη χρησιμότητά τους, καθώς και στον τρόπο αναπαράστασης της ροής τους μέσω διαγράμματος.

Διάρκεια: Μία διδακτική ώρα

3. Ενότητες 6.3, 7.1, 7.2, 7.3, 7.4

Να γίνει παραλληλισμός μεταξύ της φυσικής και της τεχνικής γλώσσας. Στη συνέχεια να γίνει παρουσίαση των συμβόλων, γραμμάτων και αριθμών που χρησιμοποιεί η ΓΛΩΣΣΑ (σύνδεση με το 6.3) και των κανόνων (γραμματικοί και συντακτικοί) που τη διέπουν. Επίσης να παρουσιασθούν, οι τύποι δεδομένων που υποστηρίζει η γλώσσα, οι μεταβλητές και οι σταθερές. Να αναλυθούν θέματα όπως: η διαφορά μεταβλητής και σταθεράς, η σχέση της μεταβλητής με τη μνήμη και οι κανόνες ονοματολογίας στις μεταβλητές. Να δοθούν παραδείγματα και ασκήσεις.

Διάρκεια: Δύο διδακτικές ώρες.

4. Ενότητες 7.5, 7.6, 7.7

Να παρουσιασθούν οι αριθμητικοί τελεστές, οι συναρτήσεις και οι μαθηματικές εκφράσεις, όπως χρησιμοποιούνται στη ΓΛΩΣΣΑ. Ιδιαίτερη έμφαση να δοθεί στη διαφορά των τελεστών $\div$ και $/$. Να παρουσιασθεί ο τρόπος γραφής μιας αριθμητικής παράστασης στον υπολογιστή, με ιδιαίτερη έμφαση στην προτεραιότητα πράξεων και στη χρήση

παρενθέσεων. Να παρουσιασθούν μαθηματικές και λοιπές βασικές συναρτήσεις σε ΓΛΩΣΣΑ. Να δοθούν παραδείγματα και ασκήσεις.

Να διευκρινιστεί ότι:

- οι συναρτήσεις $HM()$, $ΣΥΝ()$ και $ΕΦ()$ δέχονται παράμετρο σε μοίρες,
- το ακέραιο μέρος $A_M()$ ενός αριθμού χ ορίζεται όπως στα μαθηματικά ο ακέραιος με την ιδιότητα $A_M(\chi) \leq \chi < A_M(\chi) + 1$,
- η απόλυτη τιμή $A_T()$ μπορεί να πάρει ως παράμετρο, είτε ακέραιο αριθμό και να επιστρέψει ακέραιο, είτε πραγματικό αριθμό και να επιστρέψει πραγματικό.

Διάρκεια: Μία διδακτική ώρα.

5. Ενότητες 7.8, 2.4.1, 7.9, 7.10

Να παρουσιασθεί η δομή ακολουθίας (2.4.1). Να παρουσιασθούν οι εντολές εκχώρησης, εισόδου και εξόδου και οι μαθητές να δημιουργήσουν τα πρώτα προγράμματα τους με στόχο να κατανοήσουν τις εντολές. Το μάθημα να διδαχθεί στο εργαστήριο και ο καθηγητής να παρουσιάσει και έτοιμες ασκήσεις, όπου οι μαθητές μπορούν στη συνέχεια να τις εκτελέσουν στον Η/Υ. Να γίνει παρουσίαση του παραδείγματος της παραγράφου 7.10 από το Βιβλίο του Μαθητή. Είναι αποδεκτή η χρήση, είτε μονών, είτε διπλών εισαγωγικών. Να δοθούν παραδείγματα και ασκήσεις.

Διάρκεια: Μία διδακτική ώρα.

6. Ενότητες 6.4

Να διδαχθούν οι τεχνικές της ιεραρχικής σχεδίασης και του τμηματικού προγραμματισμού. Ιδιαίτερο βάρος να δοθεί στα χαρακτηριστικά και κυρίως στα πλεονεκτήματα του δομημένου προγραμματισμού. Για την εμπέδωση του μαθήματος, να δοθούν ασκήσεις θεωρητικές, απαντώντας σε ερωτήματα Σωστού-Λάθους ή ερωτήσεις ανάπτυξης.

Διάρκεια: Μία διδακτική ώρα.

7. Ενότητες 6.7

Να διδαχθούν οι έννοιες της γλώσσας υψηλού επιπέδου και της γλώσσας μηχανής, του πηγαίου και αντικείμενου προγράμματος, καθώς και αυτές του συντάκτη, των μεταφραστικών προγραμμάτων, του συνδέτη – φορτωτή και των βιβλιοθηκών. Διευκρινίζονται οι έννοιες του μεταγλωττιστή και του Διερμηνευτή και δίνεται ιδιαίτερο βάρος στις διαφορές τους, στα πλεονεκτήματα και τα μειονεκτήματά τους. Με βάση την παρουσίαση των σχημάτων της ενότητας, να περιγραφούν τα στάδια της διαδικασίας μετατροπής του πηγαίου προγράμματος σε εκτελέσιμο πρόγραμμα, με διευκρίνιση των εννοιών, που αναφέρονται στο σχήμα και ανάλυση του τρόπου λειτουργίας τους. Για την εμπέδωση του μαθήματος, να δοθούν ασκήσεις θεωρητικές, απαντώντας σε ερωτήματα Σωστού-Λάθους ή ερωτήσεις ανάπτυξης.

Διάρκεια: Μία διδακτική ώρα.

8 & 9. Ενότητες 2.4.2, 2.4.3, 2.4.4 & 8.1, 8.1.1

Να διδαχθούν, επαναληπτικά, οι λογικές πράξεις και η δομή επιλογής (απλή, πολλαπλή και εμφωλευμένη). Η εμπέδωση στις δομές αυτές προτείνεται να γίνει μέσω ημιτελών

παραδειγμάτων - ασκήσεων, τα οποία θα συμπληρώσουν οι μαθητές χωρισμένοι σε ομάδες.

Διάρκεια: Τρεις διδακτικές ώρες

Στο βιβλίο της Β' ΓΕΛ (ΕΑΕΗΥ σελ 35 στο πλαίσιο για τις Εκφράσεις, δίνεται ιεραρχία των λογικών πράξεων (1. όχι , 2. και 3. ή). Στο Βιβλίο της Γ' δεν αναφέρεται η ιεραρχία των λογικών πράξεων. Είναι δεκτή η ιεραρχία των λογικών πράξεων, όπως αναφέρεται στο βιβλίο της Β' και μπορεί να χρησιμοποιηθεί σε ασκήσεις. Προτείνεται να διδαχθεί η καλή τακτική της χρήσης παρενθέσεων.

10. Ενότητες 2.4.5, 8.2, 8.2.1

Να διδαχθεί το τμήμα της παραγράφου 2.4.5 μέχρι και το Παράδειγμα 8, εισάγοντας γενικά την έννοια της δομής επανάληψης. Να παρουσιασθεί η δομή επανάληψης ΟΣΟ ... ΕΠΑΝΑΛΑΒΕ από το 8.2.1, επισημαίνοντας σε ποιές περιπτώσεις εξυπηρετεί η χρήση της, ποιοι είναι οι βασικοί κανόνες σύνταξης της, δίνοντας ταυτόχρονα και σχετικά παραδείγματα. Να γίνει επίδειξη έτοιμου προγράμματος. Ο καθηγητής στο εργαστήριο να παρουσιάσει και έτοιμες ασκήσεις, τις οποίες οι μαθητές να τις εκτελούν στον Η/Υ.

Διάρκεια: Μία διδακτική ώρα

11. Ενότητα 8.2.2

Να παρουσιασθεί η δομή επανάληψης ΜΕΧΡΙΣ ... ΟΤΟΥ από το 8.2.2, επισημαίνοντας σε ποιές περιπτώσεις εξυπηρετεί η χρήση της, ποιοι είναι οι βασικοί κανόνες σύνταξης της, δίνοντας ταυτόχρονα και σχετικά παραδείγματα. Να διδαχθεί το Παράδειγμα 9 από την παράγραφο 2.4.5. Να παρουσιασθούν οι διαφορές και ομοιότητες ανάμεσα στις δύο πρώτες δομές επανάληψης. Να γίνει επίδειξη έτοιμου προγράμματος. Ο καθηγητής στο εργαστήριο να παρουσιάσει και έτοιμες ασκήσεις, τις οποίες οι μαθητές να τις εκτελούν στον Η/Υ.

Στο βιβλίο της Β' ΓΕΛ (ΕΑΕΗΥ Παράδειγμα 2.17) δίνεται η γενική μορφή της εντολής επανάληψης ως εξής:

Επανάλαβε

Εντολές

Μέχρις_ότου <συνθήκη>

Στο Βιβλίο της Γ' ΓΕΛ (ΑΕΠΠ) η εντολή δίνεται με την ακόλουθη σύνταξη:

Αρχή_επανάληψης

Εντολές

Μέχρις_ότου <συνθήκη>

Να διδαχθεί η σύνταξη της εντολής με τη μορφή που έχει στο βιβλίο της Γ' ΓΕΛ, αλλά σε λύσεις ασκήσεων να γίνεται δεκτή και η μορφή της εντολής που αναφέρεται στο βιβλίο της Β' ΓΕΛ.

Διάρκεια: Μία διδακτική ώρα.

12. Ενότητα 8.2.3

Να παρουσιασθεί η δομή επανάληψης ΓΙΑ ... ΑΠΟ ... ΜΕΧΡΙ από το 8.2.3, επισημαίνοντας σε ποιές περιπτώσεις εξυπηρετεί η χρήση της, ποιοι είναι οι βασικοί κανόνες σύνταξης της, δίνοντας ταυτόχρονα και σχετικά παραδείγματα. Ιδιαίτερη έμφαση να δοθεί, στο ΒΗΜΑ μεταβολής της μεταβλητής του βρόχου, δίνοντας παραδείγματα με ΒΗΜΑ αρνητικό, θετικό ή μηδέν, καθώς και στην περίπτωση όπου το ΒΗΜΑ δεν είναι υποχρεωτικό. Να διδαχθούν τα Παραδείγματα 10 και 11 από την παράγραφο 2.4.5. Να παρουσιασθούν οι κανόνες των εμφωλευμένων βρόχων. Να γίνει επίδειξη έτοιμου προγράμματος. Ο καθηγητής στο εργαστήριο να παρουσιάσει και έτοιμες ασκήσεις, τις οποίες οι μαθητές να τις εκτελούν στον Η/Υ.

Διάρκεια: Μία διδακτική ώρα.

13. Μετατροπές από μία δομή επανάληψης σε άλλη

Να παρουσιασθούν οι διαφορές και οι ομοιότητες ανάμεσα στις δομές επανάληψης, τα κύρια χαρακτηριστικά τους και σε ποιες περιπτώσεις ενδείκνυται να χρησιμοποιούμε την κάθε μία. Να διδαχθούν μετατροπές από μία δομή επανάληψης σε άλλη (βλέπε ΠΑΡΑΡΤΗΜΑ).

Διάρκεια: Δύο διδακτικές ώρες.

Στο βιβλίο της Β' (ΕΑΕΗΥ σελ 41, στο περιθώριο) Αναφέρεται:

Αν $\tau_1 > \tau_2$ και $\beta=0$ δεν θα εκτελεστούν οι εμπεριεχόμενες εντολές, ενώ αν $\tau_1 \leq \tau_2$ και $\beta=0$ θα εκτελείται άπειρες φορές (ατέρμονας βρόχος).

Στο Βιβλίο της Γ' (ΑΕΠΠ σελ 44, στην παλιά εκτύπωση του βιβλίου) αναφέρεται: "Έτσι το βήμα δεν μπορεί να είναι μηδέν γιατί τότε ο βρόχος εκτελείται επ'άπειρον".

Είναι αποδεκτή και διδάσκεται η αναφορά του βιβλίου της Γ' ΓΕΛ, δηλαδή αν το βήμα είναι μηδέν, **σε κάθε περίπτωση**, ο βρόχος εκτελείται άπειρες φορές.

15. Ενότητα 3.2

Να παρουσιασθούν οι δομές δεδομένων και οι βασικές λειτουργίες που μπορούν να εφαρμοστούν σε αυτές. Στο τέλος της παραγράφου 3.2 αναφέρονται οι στατικές και δυναμικές δομές. Να γίνει αναφορά στη διαφορά Στατικών και Δυναμικών δομών δεδομένων, σε ότι αφορά στη χρήση της μνήμης.

Διάρκεια: Μία διδακτική ώρα.

16. , 17., & 25. Ενότητες 3.3, 9.1, 9.2 & 9.4

Να παρουσιασθούν οι Στατικές δομές δεδομένων, με έμφαση στο ότι το ακριβές μέγεθος της απαιτούμενης μνήμης καθορίζεται κατά τη στιγμή του προγραμματισμού τους και ότι τα στοιχεία τους αποθηκεύονται σε συνεχόμενες θέσεις μνήμης (βλέπε ΠΑΡΑΡΤΗΜΑ). Να παρουσιασθούν οι μονοδιάστατοι πίνακες, ο τρόπος με τον οποίο ορίζονται και χρησιμοποιούνται και στη συνέχεια να διδαχθούν οι πλέον γνωστές διαδικασίες πάνω σε μονοδιάστατους πίνακες όπως, η εύρεση μεγίστου και ελαχίστου, η συγχώνευση

μονοδιάστατων πινάκων κλπ. Το μάθημα να γίνει στο εργαστήριο Πληροφορικής. Ο καθηγητής στο εργαστήριο να παρουσιάσει και έτοιμες ασκήσεις, τις οποίες οι μαθητές να τις εκτελούν στον Η/Υ. Να εξοικειωθούν οι μαθητές με το πέρασμα τιμών στη μνήμη του υπολογιστή. Να διδαχθούν παραδείγματα – ασκήσεις με εύρεση μεγίστου - ελαχίστου και αθροίσματος - μέσου όρου τιμών. Να διδαχθούν, η παράγραφος 9.1 ως έχει, χωρίς το Παράδειγμα 2, και από την 3.3 το Παράδειγμα 1 (Εύρεση του μικρότερου στοιχείου ενός μονοδιάστατου πίνακα). Να δοθεί από τον καθηγητή αντίστοιχο πρόγραμμα για την εύρεση του μεγίστου. Να διδαχθεί το Παράδειγμα 9.2, από το ΤΕΤΡΑΔΙΟ του Μαθητή και να εισαχθούν οι μαθητές την έννοια των παράλληλων πινάκων.

Διάρκεια: Τρείς διδακτικές ώρες.

18. Ενότητα 3.6

Να παρουσιασθεί η σειριακή ή γραμμική αναζήτηση σε έναν μη ταξινομημένο πίνακα. Να τονισθεί η σπουδαιότητα της χρήση μιας λογικής μεταβλητής done ως «σημαίας», προκειμένου να αποφευχθούν περιττές επαναλήψεις, Να διδαχθεί ως άσκηση η δυαδική αναζήτηση (βλέπε ΠΑΡΑΡΤΗΜΑ).

Διάρκεια: Μία διδακτική ώρα.

19. Ενότητα 3.7

Να παρουσιασθεί η έννοια της ταξινόμησης και να διδαχθεί η ταξινόμηση ευθείας ανταλλαγής. Να γίνει η επισήμανση ότι υπάρχουν διαφορετικοί αλγόριθμοι ταξινόμησης, με διαφορετική ή και ίδια πολυπλοκότητα και απόδοση (ενδεικτικά, η αναφορά σε μερικούς απλούς αλγόριθμους ταξινόμησης, στις χρήσιμες πληροφορίες στο δεξί πλαίσιο της παραγράφου 3.7). Να δοθούν, ως παραδείγματα, κάποιοι από αυτούς (ταξινόμηση με επιλογή) με μορφή ασκήσεων, όπου περιγράφεται ο αλγόριθμος και ζητείται η υλοποίηση του σε πρόγραμμα (βλέπε ΠΑΡΑΡΤΗΜΑ). Να δοθεί ιδιαίτερη προσοχή στις περιπτώσεις που υπάρχουν συνδεδεμένοι (παράλληλοι) Πίνακες. (Παράδειγμα: Ονόματα – Βαθμολογίες).

Διάρκεια: Δύο διδακτικές ώρες.

20. & 21. Ενότητες 3.4 & 3.5

Να δοθούν παραδείγματα της λειτουργίας των δύο δομών (στοίβα και ουρά), ώστε οι μαθητές να κατανοήσουν τη λειτουργία τους. Για παράδειγμα μπορεί να δοθεί το περιεχόμενο μιας δομής (στοίβας ή ουράς) και να ακολουθήσει ένα σύνολο από πράξεις εισαγωγής και εξαγωγής στοιχείων (βλέπε ΠΑΡΑΡΤΗΜΑ). Οι μαθητές θα πρέπει να είναι σε θέση να απεικονίσουν το τελικό περιεχόμενο της δομής. Σε θεωρητικές ασκήσεις μπορεί να χρησιμοποιηθούν και οι δύο δομές (ουρά και στοίβα). Ασκήσεις σε Γλώσσα (με χρήση πίνακα) μπορούν να υλοποιηθούν **μόνο** για τη δομή της στοίβας.

Διάρκεια: Δύο διδακτικές ώρες.

22. Ενότητα 3.9

Να δοθούν κατάλληλα παραδείγματα με στόχο να γνωρίσουν οι μαθητές την ύπαρξη και άλλων δομών δεδομένων (λίστες, δένδρα, γράφοι). Οι μαθητές να μπορούν να διακρίνουν το είδος της δομής, χωρίς να εμβαθύνουν στον τρόπο υλοποίησης ή λειτουργίας της (βλέπε ΠΑΡΑΡΤΗΜΑ).

Διάρκεια: Μία διδακτική ώρα.

23. & 24. Ενότητες 5.1 & 5.3

Η διαπραγμάτευση των εννοιών να γίνει με βάση το βιβλίο μαθητή και ως επιπλέον παραδείγματα να δοθούν τα παραδείγματα 1 & 2 του Κεφαλαίου 5 του τετραδίου του μαθητή. Στην παράγραφο 5.1.3 δεν προκύπτει άμεσα ότι ο χρόνος εκτέλεσης ενός προγράμματος είναι το άθροισμα των χρόνων εκτέλεσης των επιμέρους τμημάτων του και θα πρέπει να γίνει σχετική αναφορά.

Από την παράγραφο 5.3 διδάσκεται το τμήμα μέχρι τον ορισμό της πολυπλοκότητας (Μπορεί να χρησιμοποιηθεί και το υλικό της παραγράφου 2.2.3 του βιβλίου της Β' ΓΕΛ, ιδιαίτερα οι φράσεις των πλαισίων "*Η πολυπλοκότητα ενός αλγορίθμου δίνει ένα μέτρο της χρονικής καθυστέρησης του αλγορίθμου για την επίλυση ενός προβλήματος*" και "*Η πολυπλοκότητα ενός αλγορίθμου δίνει ένα μέτρο της ταχύτητας εκτέλεσης του αλγορίθμου*").

Οι μαθητές να συγκρίνουν ως προς την αποδοτικότητα τον αλγόριθμο σειριακής και δυαδικής αναζήτησης. Για τη σύγκριση αυτή, αφού βρουν το μέσο αριθμό πράξεων που απαιτεί ο αλγόριθμος σειριακής αναζήτησης η στοιχείων, να τον συγκρίνουν με τον πίνακα που δείχνει τον αριθμό των συγκρίσεων στη δυαδική αναζήτηση, για διάφορα πλήθη στοιχείων.

Για τον συμβολισμό O της πολυπλοκότητας, δεν πρέπει να αναλυθεί τι ακριβώς εκφράζει και πως υπολογίζεται σε ένα αλγόριθμο. Προτείνεται ο εκπαιδευτικός να δείξει τον πίνακα 2.2 και την εικόνα 2.10 από την παράγραφο 2.2.3 του βιβλίου της Β' ΓΕΛ, καθώς και τον πίνακα 5.4 του βιβλίου της Γ' τάξης και να συζητήσει με τους μαθητές, για την αύξηση του χρόνου ολοκλήρωσης που απαιτεί ένας αλγόριθμος, καθώς αυξάνεται η πολυπλοκότητά του.

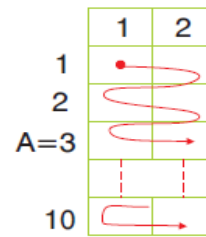
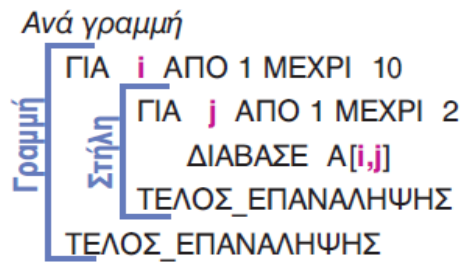
Τέλος, μπορεί να αναφερθεί ότι τα απλά προγράμματα, πρακτικά, μπορούν να αναλυθούν μετρώντας τους φωλιασμένους βρόγχους που υπάρχουν στο πρόγραμμα.

Διάρκεια: Τέσσερις διδακτικές ώρες.

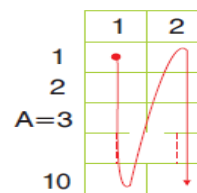
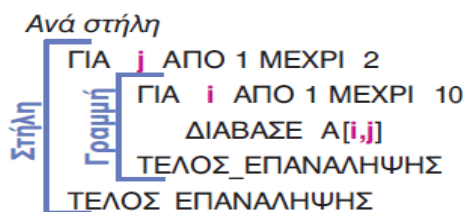
26. Ενότητες 9.3

Να παρουσιασθούν οι πολυδιάστατοι πίνακες, ο τρόπος με τον οποίο ορίζονται και χρησιμοποιούνται και τέλος να διδαχθούν οι πλέον σημαντικές διαδικασίες πάνω σε δισδιάστατους πίνακες, όπως η εύρεση μεγίστου και ελαχίστου, η αναζήτηση, η ταξινόμηση, τόσο ανά στήλη, όσο και ανά γραμμή. Η διδασκαλία να γίνει στο εργαστήριο. Να επισημανθεί ότι μπορούμε να χειριστούμε ένα δισδιάστατο πίνακα, ανάλογα με τις απαιτήσεις του προγράμματος, διαβάζοντας ή γράφοντας τα δεδομένα του πίνακα, κατά γραμμή ή κατά στήλη.

π.χ. Διάβασμα 20 ακέραιων και καταχώρηση στον πίνακα $A[10,2]$



Με τις παραπάνω εντολές γεμίζουμε τον πίνακα ανά γραμμή (όταν γεμίζει μια γραμμή, τότε συνεχίζει το γέμισμα από την αρχή της επόμενης γραμμής).



Με τις παραπάνω εντολές γεμίζουμε τον πίνακα ανά στήλη (όταν γεμίζει μια στήλη, τότε συνεχίζει το γέμισμα από την αρχή της επόμενης στήλης).

Να διδαχθεί από την παράγραφο 3.3 το ΠΑΡΑΔΕΙΓΜΑ 2 (Εύρεση αθροίσματος στοιχείων δισδιάστατου πίνακα). Να γίνει επίδειξη έτοιμων ασκήσεων από το διδάσκοντα, οι οποίες να περιέχουν τις βασικές διαδικασίες σε δισδιάστατους πίνακες (εύρεση μεγίστου – ελαχίστου, αναζήτηση στοιχείου, αθροίσματα κλπ., τόσο ανά στήλη, όσο και ανά γραμμή). Οι ασκήσεις να είναι με δισδιάστατους πίνακες και να γίνει μόνο απλή αναφορά στους πολυδιάστατους πίνακες (να δοθεί ένα παράδειγμα για το πως μπορεί να χρησιμοποιηθεί ο τρισδιάστατος πίνακας).

Διάρκεια: Τρεις διδακτικές ώρες.

28. , 29. & 30. Ενότητες 10.1, 10.2, 10.3, 10.4, 10.5, 10.6

Να παρουσιασθεί ο τμηματικός προγραμματισμός και τα πλεονεκτήματά του. Εισάγεται η έννοια του Υποπρογράμματος, ο τρόπος επικοινωνίας του με το υπόλοιπο πρόγραμμα, η λειτουργία των παραμέτρων και παρουσιάζονται οι ιδιότητες που πρέπει να το διακρίνουν.

Η διδακτική προσέγγιση να περιλαμβάνει την παρουσίαση και συζήτηση, επί ενός έτοιμου προγράμματος με υποπρογράμματα, με επίδειξη του τρόπου λειτουργίας των παραμέτρων και της εμβέλειας των μεταβλητών του: α) μέσω Διαδικασίας και β) μέσω Συνάρτησης (να αφορά στο ίδιο παράδειγμα).

Να επισημανθεί ιδιαίτερα ότι οι συναρτήσεις δεν μπορούν να έχουν εντολές εισόδου – εξόδου. Κατ' επέκταση αυτού του γεγονότος, θεωρείται, ότι δεν μπορεί να γίνει κλήση μιας διαδικασίας μέσα από μια συνάρτηση.

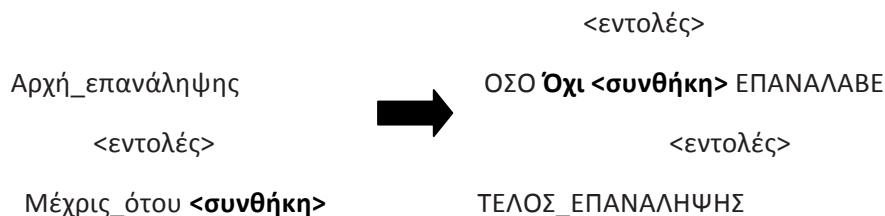
Διάρκεια: Τέσσερις διδακτικές ώρες.

ΠΑΡΑΡΤΗΜΑ

A) ΜΕΤΑΤΡΟΠΕΣ ΜΕΤΑΞΥ ΔΟΜΩΝ ΕΠΑΝΑΛΗΨΗΣ

Μετατροπή **ΑΡΧΗ ΕΠΑΝΑΛΗΨΗΣ ... ΜΕΧΡΙΣ_ΟΤΟΥ** σε **ΟΣΟ... ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ** και αντιστρόφως

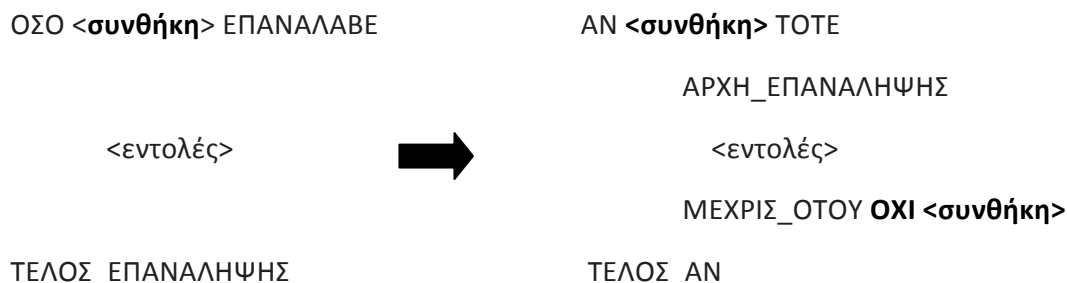
1.



- Η εντολές στη δομή επανάληψης ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ, εκτελούνται όσο η <συνθήκη> μετά το Μέχρις_ότου είναι Ψευδής, ενώ στην δομή επανάληψης ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ, εκτελούνται όσο η <συνθήκη> ανάμεσα στο ΟΣΟ και το ΕΠΑΝΑΛΑΒΕ είναι Αληθής. Γι' αυτό, κατά την μετατροπή από την μια δομή επανάληψης στην άλλη, αρκεί να γράφουμε την άρνηση της <συνθήκη> της πρώτης στη δεύτερη ή να προτάξουμε τον τελεστή ΟΧΙ στην συνθήκη.

- Επίσης, η ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ εκτελεί τουλάχιστον μια φορά όλες τις εντολές της, γι' αυτό όταν την μετατρέπουμε στην ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ πρέπει, όλες τις εντολές που περιέχει, να τις γράψουμε μια φορά πριν την ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ και άλλη μια φορά μέσα σ' αυτήν.

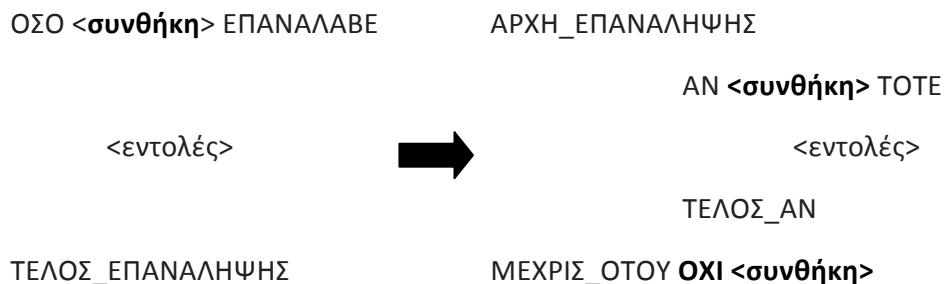
2.



- Η δομή επανάληψης ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ εκτελείται όσο η <συνθήκη> ανάμεσα στο ΟΣΟ και το ΕΠΑΝΑΛΑΒΕ είναι Αληθής, ενώ η δομή επανάληψης ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ εκτελείται όσο η συνθήκη είναι μετά το Μέχρις_ότου Ψευδής. Γι' αυτό κατά την μετατροπή από την μια δομή στην άλλη αρκεί να γράψουμε την άρνηση της <συνθήκης> της πρώτης στη δεύτερη ή να προτάξουμε τον τελεστή ΟΧΙ στην συνθήκη.

- Επίσης, η ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ μπορεί να μην εκτελεστεί καμία φορά σε αντίθεση με την ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ που θα εκτελεστεί τουλάχιστον μια φορά, γι' αυτό πριν την δομή ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ θα χρησιμοποιηθεί μια εντολή ελέγχου, που αν ισχύει η <συνθήκη> θα εκτελεστεί η ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ.

*Ως μη βέλτιστη λύση (η συνθήκη ελέγχεται δύο φορές), για την ίδια μετατροπή μπορεί να δοθεί και η παρακάτω:



Μετατροπή από ΓΙΑ... σε ΟΣΟ... ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Περίπτωση $\text{τιμή1} \leq \text{τιμή2}$ και $\beta > 0$	Περίπτωση $\text{τιμή1} \geq \text{τιμή2}$ και $\beta < 0$
ΓΙΑ <μεταβλητή> ΑΠΟ τιμή1 ΜΕΧΡΙ τιμή2 ΜΕ_ΒΗΜΑ β <εντολές> ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ↓ <μεταβλητή> $\leftarrow$ τιμή1 ΟΣΟ <μεταβλητή> $\leq$ τιμή2 ΕΠΑΝΑΛΑΒΕ <εντολές> <μεταβλητή> $\leftarrow$ <μεταβλητή> + β ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	ΓΙΑ <μεταβλητή> ΑΠΟ τιμή1 ΜΕΧΡΙ τιμή2 ΜΕ_ΒΗΜΑ β <εντολές> ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ↓ <μεταβλητή> $\leftarrow$ τιμή1 ΟΣΟ <μεταβλητή> $\geq$ τιμή2 ΕΠΑΝΑΛΑΒΕ <εντολές> <μεταβλητή> $\leftarrow$ <μεταβλητή> + β ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

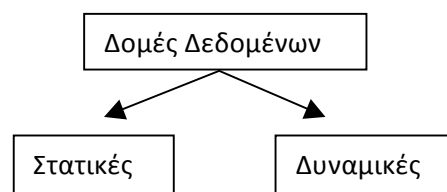
Πριν την εντολή ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ εκχωρούμε στην <μεταβλητή> της ΓΙΑ... την αρχική τιμή δηλ. την τιμή1.

- Στη συνθήκη της ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ συγκρίνουμε την <μεταβλητή> με την τιμή2. Η σύγκριση εξαρτάται από το βήμα αν είναι θετικό ή αρνητικό όπως βλέπουμε παραπάνω.

- Πριν το ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ της ΟΣΟ... αυξάνουμε την τιμή της μεταβλητής όσο είναι η τιμή του βήματος. Στη περίπτωση που το βήμα δεν υπάρχει, τότε η <μεταβλητή> αυξάνεται κατά 1.

- Η μετατροπή της ΟΣΟ... σε ΓΙΑ..., γίνεται μόνο στην περίπτωση που στην ΟΣΟ... είναι γνωστός ο αριθμός των επαναλήψεων, σε οποιαδήποτε άλλη περίπτωση δεν μετατρέπεται η ΟΣΟ... σε ΓΙΑ....

Β) ΣΤΑΤΙΚΕΣ ΚΑΙ ΔΥΝΑΜΙΚΕΣ ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ



Χαρακτηριστικά των Στατικών και Δυναμικών δομών δεδομένων

Στατικές δομές: Αποθηκεύονται σε συνεχόμενες θέσεις μνήμης και έχουν σταθερό μέγεθος, το οποίο καθορίζεται στην αρχή του προγράμματος. Οι στατικές δομές υλοποιούνται με πίνακες.

Δυναμικές δομές: Δεν αποθηκεύονται σε συνεχόμενες θέσεις μνήμης, δεν έχουν σταθερό μέγεθος, αλλά ο αριθμός των κόμβων τους αυξάνεται και μειώνεται, όταν στη δομή αντίστοιχα εισάγονται ή διαγράφονται δεδομένα. Το μέγεθος της μνήμης καθορίζεται κατά την στιγμή της εκτέλεσης του προγράμματος. Με δυναμικές δομές υλοποιούνται οι λίστες, τα δένδρα και οι γράφοι.

Πέρα από την διαφορά τους στην αποθήκευση στην κύρια μνήμη, οι μαθητές θα πρέπει να κατανοήσουν ότι για τις στατικές δομές (όπως αντιμετωπίζονται και στο βιβλίο) πρέπει να ορίζουν το μέγεθός τους, πριν από την έναρξη του προγράμματος, στο τμήμα δηλώσεων. Αντίθετα, για τις δυναμικές δομές μπορούμε να ορίζουμε και να τροποποιούμε το μέγεθος τους μέσα από το πρόγραμμα.

Πρέπει να τονιστεί, ότι μια δομή δεδομένων δεν είναι εγγενώς στατική ή δυναμική, αλλά εξαρτάται από τις δυνατότητες της γλώσσας προγραμματισμού που χρησιμοποιούμε και από τον τρόπο υλοποίησης της δομής στη γλώσσα αυτή. Οποιαδήποτε γλώσσα προγραμματισμού δεν υποστηρίζει όλες τις δομές δεδομένων, με τις σύγχρονες γλώσσες προγραμματισμού να υποστηρίζουν δυναμικές δομές δεδομένων.

Η γλώσσα προγραμματισμού ΓΛΩΣΣΑ, που χρησιμοποιείται στο βιβλίο, υποστηρίζει μόνο στατικές δομές.

Να σημειωθεί ότι οι πράξεις, των δομών της παραγράφου 3.2, αναφέρονται γενικά και αποκτούν πιο συγκεκριμένη σημασία, ανάλογα με τη δομή στην οποία αναφερόμαστε. Για παράδειγμα σε μια δομή πίνακα, κατά την πράξη της ταξινόμησης, δεν αναδιατάσσονται οι κόμβοι του αλλά το περιεχόμενο των κόμβων.

Επισημαίνεται ότι οι πίνακες στο βιβλίο της Β' τάξης αντιμετωπίζονται ως δυναμικές δομές, ενώ στο βιβλίο της Γ' τάξης ορίζονται ως στατικές δομές. Συνεπώς για τη Γ' τάξη και τη ΓΛΩΣΣΑ, η δομή του πίνακα είναι στατική και για να χρησιμοποιηθεί ένας πίνακας θα πρέπει να έχει πρώτα δηλωθεί, τόσο ο πίνακας, όσο και το μέγεθός του. Επίσης και οι δομές ουρά και στοίβα θεωρούνται στατικές δομές για τη ΓΛΩΣΣΑ, επειδή υλοποιούνται με πίνακες.

Γ) ΔΥΑΔΙΚΗ ΑΝΑΖΗΤΗΣΗ

Ο αλγόριθμος της δυαδικής αναζήτησης (binary search) εφαρμόζεται μόνο σε πίνακες που έχουν ταξινομημένα στοιχεία. Αν τα στοιχεία δεν είναι ταξινομημένα τότε δεν μπορεί να εφαρμοστεί.

Ο αλγόριθμος λειτουργεί ως εξής:

Βρίσκουμε το μεσαίο στοιχείο του ταξινομημένου πίνακα.

Εάν το προς αναζήτηση στοιχείο είναι ίσο με το μεσαίο στοιχείο τότε σταματάμε την αναζήτηση αφού το στοιχείο βρέθηκε

Εάν δεν βρέθηκε, τότε ελέγχουμε αν το στοιχείο που αναζητούμε είναι μικρότερο ή μεγαλύτερο από το μεσαίο στοιχείο του πίνακα. Αν είναι μικρότερο, περιορίζουμε την αναζήτηση στο πρώτο μισό του πίνακα (με την προϋπόθεση ότι τα στοιχεία είναι διατεταγμένα κατά αύξουσα σειρά), ενώ αν είναι μεγαλύτερο περιορίζουμε την αναζήτηση στο δεύτερο μισό του πίνακα.

Η διαδικασία αυτή λοιπόν επαναλαμβάνεται για το κατάλληλο πρώτο ή δεύτερο μισό πίνακα, μετά για το 1/4 του πίνακα κ.ο.κ. μέχρι, είτε να βρεθεί το στοιχείο, είτε να μην είναι δυνατό να χωρισθεί ο πίνακας περαιτέρω σε δύο νέα μέρη.

Αλγόριθμος δυαδικής αναζήτησης

αλγόριθμος Δυαδική_αναζήτηση !Α μονοδιάστατος πίνακας N θέσεων, S το στοιχείο που αναζητούμε

δεδομένα // N, A, S //

Left $\leftarrow$ 1 ! αριστερό όριο

Right $\leftarrow$ N ! δεξιό όριο

K $\leftarrow$ 0 ! θέση του στοιχείου

F $\leftarrow$ FALSE

όσο (Left $\leq$ Right) και (f=FALSE) **επανάλαβε**

M $\leftarrow$ (Left+Right) div 2

αν A[M]=S **τότε**

K $\leftarrow$ M;

F $\leftarrow$ TRUE;

```

αλλιώς
    αν  $A[M] < S$  τότε
        Left  $\leftarrow M+1$ ;
    αλλιώς
        Right  $\leftarrow M-1$ ;
    Τέλος_αν
Τέλος_αν
Τέλος_επανάληψης
Αν  $F = \text{TRUE}$  τότε
    Εμφάνισε "Το στοιχείο", S, "υπάρχει στη θέση:", M
Αλλιώς
    Εμφάνισε "Το στοιχείο", S, "δεν υπάρχει στον πίνακα"

```

Τέλος_αν

Η δυαδική αναζήτηση να διδαχθεί ως άσκηση και να υλοποιηθεί με πρόγραμμα, όπως παρακάτω σε ταξινομημένο πίνακα 20 θέσεων. Πέρα από το τμήμα δηλώσεων, το πρόγραμμα έχει ένα επιπλέον τμήμα για το "γέμισμα" του πίνακα με στοιχεία (υποθέτουμε ότι ο πίνακας γεμίζει με σωστά ταξινομημένα στοιχεία σε αύξουσα σειρά).

ΠΡΟΓΡΑΜΜΑ δυαδική_αναζήτηση

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[20], Left, Right, M, k, S, i

ΛΟΓΙΚΕΣ: f

ΑΡΧΗ

ΓΡΑΨΕ 'Οι αριθμοί που θα δοθούν πρέπει να είναι ταξινομημένοι κατά αύξουσα τάξη'

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 20

ΓΡΑΨΕ 'Δώσε το', i, 'στοιχείο του πίνακα'

ΔΙΑΒΑΣΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Δώσε τιμή για αναζήτηση: '

ΔΙΑΒΑΣΕ S

Left $\leftarrow$ 1

Right $\leftarrow$ 20

k $\leftarrow$ 0

f $\leftarrow$ ΨΕΥΔΗΣ

ΟΣΟ (Left $\leq$ Right) **ΚΑΙ** (f = ΨΕΥΔΗΣ) **ΕΠΑΝΑΛΑΒΕ**

 M $\leftarrow$ (Left + Right) DIV 2

ΑΝ A[M] = S **ΤΟΤΕ**

 k $\leftarrow$ M

 f $\leftarrow$ ΑΛΗΘΗΣ

ΑΛΛΙΩΣ

ΑΝ $A[M] < S$ ΤΟΤΕ

Left $\leftarrow M + 1$

ΑΛΛΙΩΣ

Right $\leftarrow M - 1$

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ $f = \text{ΑΛΗΘΗΣ}$ ΤΟΤΕ

ΓΡΑΨΕ "Το στοιχείο, ", S, " υπάρχει στη θέση:", M

ΑΛΛΙΩΣ

ΓΡΑΨΕ "Το στοιχείο, ", S, " δεν υπάρχει στον πίνακα"

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ δυαδική_αναζήτηση

Παράδειγμα

Δίνεται ο πίνακας

1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
---	---	---	---	---	----	----	----	----	----	----	----	----	----	----

Αναζήτηση του στοιχείου 38 (υπάρχει στον πίνακα)

Βήμα 1	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 2	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 3	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 4	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47

Με κίτρινο σημειώνεται το στοιχείο του πίνακα που εξετάζεται (στο μέσον)

Με πράσινο σημειώνεται το τμήμα του πίνακα που απομένει για αναζήτηση

Με κόκκινο σημειώνεται το τμήμα του πίνακα που έχει αποκλειστεί

(Τα ίδια χρώματα χρησιμοποιούνται και στο επόμενο παράδειγμα)

Αναζήτηση του στοιχείου 39 (δεν υπάρχει στον πίνακα)

Βήμα 1	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 2	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 3	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 4	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47
Βήμα 5	1	2	5	8	9	15	22	27	35	37	38	40	43	45	47

Αριθμός συγκρίσεων στη δυαδική αναζήτηση

Στοιχεία N	Συγκρίσεις
10	4
100	7
1.000	10
10.000	14
100.000	17
1.000.000	20
10.000.000	24
100.000.000	27
1.000.000.000	30

*Ως άσκηση μπορεί να δοθεί η βελτιστοποίηση του αλγορίθμου δυαδικής αναζήτησης έτσι ώστε να επιτρέπει διαδοχικές αναζητήσεις πολλών στοιχείων. Η αναζήτηση να τερματίζεται όταν δοθεί κάποιος συγκεκριμένος αριθμός ή με ερώτηση "Θέλετε άλλη αναζήτηση (N/O)"

Δ) ΤΑΞΙΝΟΜΗΣΗ ΜΕ ΕΠΙΛΟΓΗ (SELECTION SORT)

Η ταξινόμηση με επιλογή (selection sort), αποτελεί βασικό τρόπο ταξινόμησης, που υλοποιείται σε ένα μονοδιάστατο πίνακα σε τρία βήματα:

1. Επιλογή του ελάχιστου στοιχείου
2. Ανταλλαγή του ελάχιστου με το πρώτο στοιχείο
3. Επανάληψη των βημάτων 1 και 2 για τα υπόλοιπα στοιχεία του πίνακα

Ο Αλγόριθμος ταξινόμησης με επιλογή είναι ο παρακάτω.

Αλγόριθμος Selection_Sort

Δεδομένα // table, n //

Για i από 1 μέχρι n-1

$k \leftarrow i$

$x \leftarrow \text{table}[i]$

Για j από i+1 μέχρι n

Αν $x > \text{table}[j]$ Τότε

$k \leftarrow j$

$x \leftarrow \text{table}[j]$

Τέλος_Επανάληψης

$\text{table}[k] \leftarrow \text{table}[i]$

$\text{table}[i] \leftarrow x$

Τέλος_επανάληψης

Η υλοποίηση του αλγορίθμου ταξινόμησης με επιλογή, να διδαχθεί ως άσκηση και να υλοποιηθεί με πρόγραμμα όπως παρακάτω. Πέρα από το τμήμα δηλώσεων, το πρόγραμμα έχει δύο επιπλέον τμήματα, ένα τμήμα για το "γέμισμα" του πίνακα με στοιχεία και ένα τμήμα για την εκτύπωση του ταξινομημένου πίνακα.

ΠΡΟΓΡΑΜΜΑ Selection_Sort

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[20], K1, x, i, j

ΑΡΧΗ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 20

ΓΡΑΨΕ 'Δώσε το', i, ' στοιχείο του πίνακα'

ΔΙΑΒΑΣΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 19

K1 <- i

x <- A[i]

ΓΙΑ j **ΑΠΟ** i + 1 **ΜΕΧΡΙ** 20

ΑΝ x > A[j] **ΤΟΤΕ**

K1 <- j

x <- A[j]

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

A[K1] <- A[i]

A[i] <- x

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Εκτύπωση με ταξινομημένα τα στοιχεία'

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 20

ΓΡΑΨΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Selection_Sort

Παράδειγμα

Αν υποθέσουμε ότι έχουμε το πίνακα A[8] με στοιχεία τους αριθμούς 46, 55, 12, 42, 94, 18, 06, 67. Δηλαδή σε μορφή μονοδιάστατου πίνακα:

46	55	12	42	94	18	06	67
----	----	----	----	----	----	----	----

τότε παρακάτω φαίνεται πως μετακινούνται τα στοιχεία με τον αλγόριθμο SelectionSort

Βήμα 1 (εύρεση του ελάχιστου των στοιχείων και ανταλλαγή με το πρώτο)

46	55	12	42	94	18	06	67
----	----	----	----	----	----	----	----

Βήμα 2 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το δεύτερο στοιχείο και κάτω)

06	55	12	42	94	18	46	67
----	----	----	----	----	----	----	----

Βήμα 3 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το τρίτο στοιχείο και κάτω)

06	12	55	42	94	18	46	67
----	----	----	----	----	----	----	----

Βήμα 4 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το τέταρτο στοιχείο και κάτω)

06	12	18	42	94	55	46	67
----	----	----	----	----	----	----	----

Βήμα 5 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το πέμπτο στοιχείο και κάτω)

06	12	18	42	94	55	46	67
----	----	----	----	----	----	----	----

Βήμα 6 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το έκτο στοιχείο και κάτω)

06	12	18	42	46	55	94	67
----	----	----	----	----	----	----	----

Βήμα 7 (επανάληψη της ανωτέρω διαδικασίας αλλά στο τμήμα του πίνακα από το έβδομο στοιχείο και κάτω)

06	12	18	42	46	55	94	67
----	----	----	----	----	----	----	----

Τελική μορφή ταξινομημένου πίνακα (δεν χρειάζεται 8^η επανάληψη σύγκρισης, αφού όταν απομένουν δύο μόνο κελιά και στο πρώτο θέσεις τον μικρότερο αριθμό, τότε στο δεύτερο αναγκαστικά τίθεται ο μεγαλύτερος)

06	12	18	42	46	55	67	94
----	----	----	----	----	----	----	----

Ε) ΠΡΟΤΕΙΝΟΜΕΝΕΣ ΘΕΩΡΗΤΙΚΕΣ ΑΣΚΗΣΕΙΣ ΣΕ ΣΤΟΙΒΑ ΚΑΙ ΟΥΡΑ

- 1) Σε μια στοίβα 10 θέσεων έχουν τοποθετηθεί διαδοχικά τα στοιχεία: Σ, Γ, Μ, Α, Δ στην 1η, 2η, 3η, 4η και 5η θέση αντίστοιχα.
 - i) Να προσδιορίσετε την τιμή του δείκτη **top** της παραπάνω στοίβας και να την σχεδιάσετε.
 - ii) Αν εφαρμόσουμε τις παρακάτω λειτουργίες: **Απώθηση**, **Απώθηση**, **Απώθηση**, **Ώθηση Χ**, **Ώθηση Δ** και **Απώθηση** ποιά είναι η νέα τιμή της **top** και ποιά η τελική μορφή της στοίβας;
- 2) Η παραπάνω άσκηση να υλοποιηθεί με ουρά χρησιμοποιώντας, όπου Απώθηση **Εξαγωγή** και όπου Ώθηση **Εισαγωγή**. Επίσης αντί της top να δοθούν οι τιμές των δεικτών rear και front.
- 3) Σε μια άδεια στοίβα 10 θέσεων ωθούμε τα στοιχεία Ο, Σ, Ε, Τ, Λ. Με ποίο τρόπο πρέπει να ωθηθούν και να απωθηθούν τα δεδομένα ώστε η στοίβα να περιέχει τα δεδομένα Τ, Ε, Λ, Ο, Σ (σε αύξουσες θέσεις του πίνακα).
- 4) Η παραπάνω άσκηση να υλοποιηθεί με ουρά, χρησιμοποιώντας όπου Απώθηση **Εξαγωγή** και όπου Ώθηση **Εισαγωγή**. Επίσης αντί της top να δοθούν οι τιμές των δεικτών rear και front.

Η αρχική μορφή της στοίβας είναι:

10	
9	
8	
7	
6	
5	Λ
4	Τ
3	Ε
2	Σ
1	Ο

top →

Εκτελώντας τις λειτουργίες : **Απώθηση, Απώθηση, Απώθηση, Απώθηση, Ώθηση Τ, Ώθηση Ε, Ώθηση Λ, Ώθηση Ο, Ώθηση Σ**, τότε η τελική μορφή της στοίβας γίνεται:

10	
9	
8	
7	
6	
5	Σ
4	Ο
3	Λ
2	Ε
1	Τ

top →

4) i) Η αρχική μορφή της ουράς είναι:

1η	2η	3η	4η	5η	6η	7η	8η	9η	10η
Ο	Σ	Ε	Τ	Λ					
front		rear							

οι τιμές της front και της rear είναι: front=1 και rear=5

ii) Εκτελώντας τις λειτουργίες : **Εξαγωγή, Εξαγωγή, Εξαγωγή, Εξαγωγή, Εξαγωγή, Εισαγωγή Τ, Εισαγωγή Ε, Εισαγωγή Λ, Εισαγωγή Ο, Εισαγωγή Σ**, τότε η τελική μορφή της ουράς γίνεται

1η	2η	3η	4η	5η	6η	7η	8η	9η	10η
					Τ	Ε	Λ	Ο	Σ
front					rear				

και οι τιμές της front και της rear γίνονται: front=6 και rear=10.

Επίσης προτείνονται προς διδασκαλία και οι ακόλουθες ασκήσεις της ίδιας κατηγορίας.

1) Σε μια κενή στοίβα πρόκειται να εισαχθούν τα στοιχεία Α, Μ, D, Κ, L, Β με τη σειρά που δίνονται (Α πρώτο, Β τελευταίο). Ακολουθεί μια σειρά πράξεων που είναι:

α) Ώθηση δύο στοιχείων στη στοίβα και απώθηση ενός

- β) Ώθηση δύο στοιχειών στη στοίβα και απώθηση ενός
 γ) Ώθηση δύο στοιχειών στη στοίβα και απώθηση ενός
 Ποια στοιχεία και με ποια σειρά, περιέχει η στοίβα μετά τις πράξεις αυτές;
 Η ανωτέρω άσκηση μπορεί να υλοποιηθεί σε γλώσσα με χρήση ενός πίνακα 10 θέσεων
- 2) Σε μια κενή ουρά πρόκειται να εισαχθούν τα στοιχεία A, M, D, K, L, B με τη σειρά που δίνονται (A πρώτο, B τελευταίο). Ακολουθεί μια σειρά πράξεων που είναι:
- α) Εισαγωγή δύο στοιχειών στη στοίβα και εξαγωγή ενός
 β) Εισαγωγή δύο στοιχειών στη στοίβα και εξαγωγή ενός
 γ) Εισαγωγή δύο στοιχειών στη στοίβα και εξαγωγή ενός
 Ποια στοιχεία και με ποια σειρά, περιέχει η ουρά μετά τις πράξεις αυτές;

Υλοποίηση σε γλώσσα ώθησης και απώθησης σε στοίβα με χρήση πίνακα

1) Το τμήμα προγράμματος για την ώθηση σε στοίβα είναι το παρακάτω: ΓΡΑΨΕ 'Δώσε στοιχείο για να εισαχθεί στη στοίβα A:' ΔΙΑΒΑΣΕ στοιχείο AN top<10 TOTE top <-- top + 1 A[top]<-- στοιχείο ΑΛΛΙΩΣ ΓΡΑΨΕ 'Υπερχείλιση στοίβας' ΤΕΛΟΣ_ΑΝ	2) Το τμήμα προγράμματος για την απώθηση από στοίβα είναι το παρακάτω: AN top>=1 TOTE Στοιχείο <-- A[top] top<-- top-1 ΑΛΛΙΩΣ ΓΡΑΨΕ 'Υποχείλιση στοίβας' ΤΕΛΟΣ_ΑΝ
--	--

Επισήμανση: Τα παραπάνω τμήματα προγράμματος μπορεί να αναφερθούν κατά τη διδασκαλία της ενότητας των υποπρογραμμάτων ως διαδικασίες, όπως φαίνεται παρακάτω.

ΥΠΟΠΡΟΓΡΑΜΜΑΤΑ ΓΙΑ ΩΘΗΣΗ ΚΑΙ ΑΠΩΘΗΣΗ ΣΕ ΣΤΟΙΒΑ

Στα υποπρογράμματα που ακολουθούν, το στοιχείο που ωθείται ή απωθείται στη στοίβα, είναι παράμετρος στις διαδικασίες (για είσοδο στην ώθηση ή έξοδο από την απώθηση) και υπάρχει και η λογική μεταβλητή done που επιστρέφει τιμή ΑΛΗΘΗΣ ή ΨΕΥΔΗΣ, αναλόγως αν έγινε η ώθηση ή η απώθηση στη στοίβα. Το μέγεθος N του πίνακα έχει δηλωθεί ως συμβολική σταθερά για να μπορεί να αλλαχθεί ανάλογα με το πρόβλημα.

ΔΙΑΔΙΚΑΣΙΑ ΩΘΗΣΗ (A, στοιχείο, top, done)	ΔΙΑΔΙΚΑΣΙΑ ΑΠΩΘΗΣΗ (A, στοιχείο, top, done)
ΣΤΑΘΕΡΕΣ	ΣΤΑΘΕΡΕΣ
N = 10	N = 10
ΜΕΤΑΒΛΗΤΕΣ	ΜΕΤΑΒΛΗΤΕΣ
ΑΚΕΡΑΙΕΣ: top	ΑΚΕΡΑΙΕΣ: top
ΧΑΡΑΚΤΗΡΕΣ: στοιχείο, A[N]	ΧΑΡΑΚΤΗΡΕΣ: στοιχείο, A[N]
ΛΟΓΙΚΕΣ: done	ΛΟΓΙΚΕΣ: done
ΑΡΧΗ	ΑΡΧΗ
ΑΝ top < N ΤΟΤΕ	ΑΝ top >= 1 ΤΟΤΕ
top <-- top + 1	στοιχείο <-- A[top]
A[top] <-- στοιχείο	top <-- top - 1
done <-- ΑΛΗΘΗΣ	done <-- ΑΛΗΘΗΣ
ΑΛΛΙΩΣ	ΑΛΛΙΩΣ
done <-- ΨΕΥΔΗΣ	done <-- ΨΕΥΔΗΣ
ΤΕΛΟΣ_ΑΝ	ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ	ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ

Υλοποίηση σε γλώσσα στοίβας με χρήση πίνακα.

Άσκηση

Ένα οχηματαγωγό πλοίο, χωρητικότητας 250 αυτοκινήτων, εκτελεί το δρομολόγιο ΠΕΙΡΑΙΑΣ – ΑΙΓΙΝΑ. Τα οχήματα που επιβιβάζονται πρώτα είναι αυτά που θα αποβιβαστούν τελευταία. Στο λιμάνι του Πειραιά προσέρχονται τα αυτοκίνητα για αναχώρηση. Να γίνει πρόγραμμα το οποίο:

1. Να υπάρχει μενού επιλογής:
 1. Επιβίβαση
 2. Αποβίβαση
 3. Έξοδος
2. Στη περίπτωση που επιλεγθεί η Επιβίβαση θα διαβάζει τον αριθμό κυκλοφορίας καθενός από τα αυτοκίνητα που προσέρχονται και ο αριθμός κυκλοφορίας του να καταχωρείται στη στοίβα ΟΧΗΜΑΤΑ. Κάθε φορά που επιβιβάζεται ένα αυτοκίνητο να τυπώνεται το ερώτημα "Υπάρχει άλλο αυτοκίνητο (N/O); ". Αν ο χρήστης απαντήσει N (=ΝΑΙ), επαναλαμβάνεται η διαδικασία επιβίβασης, ενώ αν απαντήσει Ο (=ΟΧΙ), σταματά η διαδικασία επιβίβασης και επιστρέφει το πρόγραμμα στο μενού Επιλογής.
3. Αν το πλοίο γεμίσει η επιβίβαση σταματά εμφανίζεται κατάλληλο μήνυμα και επιστρέφει το πρόγραμμα στο μενού επιλογής.
4. Στη περίπτωση που επιλεγθεί η **Αποβίβαση**, εξάγει και εμφανίζει από την στοίβα ΟΧΗΜΑΤΑ όλους τους αριθμούς αυτοκινήτων που είχαν επιβιβαστεί

στον ΠΕΙΡΑΙΑ, με τη σειρά που αποβιβάζονται. Στο τέλος να τυπώνεται το πλήθος των αυτοκινήτων που αποβιβάστηκαν στο λιμάνι της ΑΙΓΙΝΑΣ

Απάντηση

ΠΡΟΓΡΑΜΜΑ Λιμάνι

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: τοπ, επ1, πλ

ΧΑΡΑΚΤΗΡΕΣ: επ2, αρ, π[5]

ΑΡΧΗ

τοπ <- 0

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ ' Μενού Επιλογών'

ΓΡΑΨΕ ' 1. Επιβίβαση'

ΓΡΑΨΕ ' 2. Αποβίβαση'

ΓΡΑΨΕ ' 3. Έξοδος'

ΓΡΑΨΕ ' Δώσε επιλογή:'

ΔΙΑΒΑΣΕ επ1

ΑΝ επ1 = 1 **ΤΟΤΕ**

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ ' Υπάρχει αυτοκίνητο για επιβίβαση (Ν/Ο);'

ΔΙΑΒΑΣΕ επ2

ΑΝ επ2 <> 'Ν' **ΚΑΙ** επ2 <> 'ν' **ΚΑΙ** επ2 <> 'Ο' **ΚΑΙ** επ2 <> 'ο' **ΤΟΤΕ**

ΓΡΑΨΕ 'Λάθος επιλογή. Ξαναπροσπάθησε!!!'

ΤΕΛΟΣ_ΑΝ

ΜΕΧΡΙΣ_ΟΤΟΥ επ2 = 'Ο' **Η** επ2 = 'ο' **Η** επ2 = 'Ν' **Η** επ2 = 'ν'

ΑΝ επ2 = 'Ν' **Η** επ2 = 'ν' **ΤΟΤΕ**

ΑΝ τοπ < 5 **ΤΟΤΕ**

ΓΡΑΨΕ 'Δώσε αριθμό κυκλοφορίας του αυτοκινήτου:'

ΔΙΑΒΑΣΕ αρ

τοπ <- τοπ + 1

π[τοπ] <- αρ

ΑΝ τοπ = 5 **ΤΟΤΕ**

ΓΡΑΨΕ 'Το πλοίο γέμισε και δεν χωρά άλλα αμάξια'

ΤΕΛΟΣ_ΑΝ

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Το πλοίο είναι γεμάτο'

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΜΕΧΡΙΣ_ΟΤΟΥ τοπ = 5 **Η** επ2 = 'Ο' **Η** επ2 = 'ο'

ΑΛΛΙΩΣ_ΑΝ επ1 = 2 **ΤΟΤΕ**

πλ <- 0

ΟΣΟ τοπ >= 1 **ΕΠΑΝΑΛΑΒΕ**

ΓΡΑΨΕ 'Αποβιβάζεται το αυτοκίνητο με αριθμό κυκλοφορίας:', π[τοπ]

π[τοπ] <- ' '

τοπ <- τοπ - 1

πλ <- πλ + 1

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Πλήθος οχημάτων που αποβιβάστηκαν στο λιμάνι της ΑΙΓΙΝΑΣ:', πλ
ΤΕΛΟΣ_ΑΝ
ΜΕΧΡΙΣ_ΟΤΟΥ επ1 = 3
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Λιμάνι

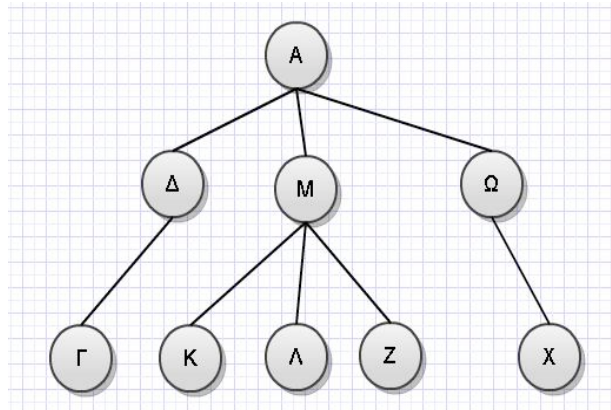
ΣΤ) ΑΛΛΕΣ ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

ΔΕΝΤΡΑ (TREES)

Η έννοια του δέντρου είναι στενά συνδεδεμένη με την έννοια της ιεραρχίας και αποτελεί μια μη γραμμική δομή.

Το Δέντρο είναι ένα πεπερασμένο σύνολο κόμβων (ίδιου τύπου) και ακμών που συνδέουν τους κόμβους, με βάση κάποια σχέση που δημιουργεί την ιεραρχική δομή των κόμβων.

Ένας από τους κόμβους αποτελεί και τη ρίζα του δέντρου. Στο ανωτέρω σχήμα ρίζα είναι ο κόμβος Α.



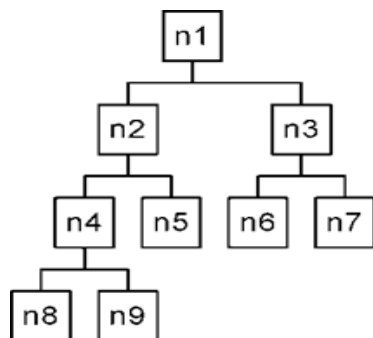
Όταν αναφερόμαστε σε μια δομή δέντρου, χρησιμοποιούμε πολύ συχνά τους όρους **γονέας** και **παιδιά** (ο κόμβος Α είναι γονέας των κόμβων Δ, Μ και Ω, ενώ τα Δ, Μ και Ω λέγονται παιδιά του Α).

Οι κόμβοι ενός δέντρου, από τους οποίους δεν αρχίζει κάποιο υποδέντρο (δεν έχουν δηλαδή παιδιά), ονομάζονται **φύλλα** (leaves). Όλοι οι άλλοι κόμβοι ονομάζονται μη-τερματικοί ή **κλαδιά** (branches).

Υπάρχουν πολλά είδη δέντρων. Ένα από τα πιο γνωστά είναι το ονομαζόμενο **δυναμικό δέντρο** (Binary tree), όπου κάθε μη τερματικός κόμβος έχει ακριβώς δύο παιδιά.

Ο αριθμός των παιδιών ενός κόμβου ορίζει το **βαθμό** (degree) του κόμβου. Ο βαθμός ενός δέντρου είναι ο μέγιστος βαθμός από όλους τους βαθμούς των κόμβων του. Τα δυναμικά δέντρα έχουν βαθμό 2.

Επίπεδο (level) ενός κόμβου είναι το μήκος της μοναδικής διαδρομής από την ρίζα προς αυτόν τον κόμβο. Η ρίζα κάθε δέντρου βρίσκεται στο μηδενικό επίπεδο.



Έννοιες των δέντρων με βάση το σχήμα

Διαδρομή από το n2 προς το n9 είναι η ακολουθία n2, n4, n9. Το **μήκος** της διαδρομής αυτής είναι 2.

Φύλλα είναι τα n8, n9, n5, n6 και n7.

Το **ύψος** του δέντρου είναι 3, ενώ το ύψος του κόμβου n2 είναι 2 και του n9 μηδέν.

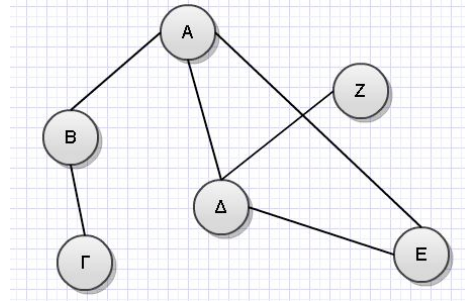
Ο **βαθμός** του κόμβου n9 είναι μηδέν, ενώ του κόμβου n2 είναι 2.

Το **επίπεδο** του κόμβου n2 είναι ένα και του n9 είναι τρία.

ΓΡΑΦΟΙ

Η δομή του γράφου (graph) είναι η πιο γενική μορφή δομής δεδομένων. Αυτό σημαίνει ότι οι δομές που εξετάσαμε προηγουμένως μπορούν να θεωρηθούν ως υποπεριπτώσεις των γράφων.

Το γενικότερο χαρακτηριστικό τους είναι ότι δεν υπάρχει κάποια ιεραρχική δομή και κάθε κόμβος μπορεί να συνδέεται με οποιονδήποτε άλλον.



Οι διδάσκοντες να ενημερωθούν ενυπόγραφα.

**Ο ΥΠΟΥΡΓΟΣ
ΠΑΙΔΕΙΑΣ, ΕΡΕΥΝΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ**

ΝΙΚΟΛΑΟΣ ΦΙΛΗΣ

Εσωτ. Διανομή

- Δ/ση Σπουδών, Προγρ/των & Οργάνωσης Δ.Ε., Τμ. Α΄
- Αυτ. Δ/ση Παιδείας, Ομογ., Διαπολ. Εκπ/σης, Ξένων και Μειον. Σχολείων
- Διεύθυνση Θρησκευτικής Εκπ/σης
- Δ/ση Ειδικής Αγωγής και Εκπ/σης
- Διεύθυνση Εξετάσεων και Πιστοποιήσεων, Τμ. Α΄